Comment on “New Results on Frame-Proof Codes and Traceability Schemes”

Jan-Åke Larsson and Jacob Löfvenberg

***Abstract*— In the paper “New Results on Frame-Proof Codes and Traceability Schemes” by Reihaneh Safavi-Naini and Yejing Wang [IEEE Trans. Inform. Theory, vol. 47, no. 7, pp. 3029–3033, Nov. 2001], there are lower bounds for the maximal number of codewords in binary frame-proof codes and decoders in traceability schemes. There are also existence proofs using a construction of binary frame-proof codes and traceability schemes. Here it is found that the main results in the referenced paper do not hold.**

***Index Terms*— fingerprinting, watermarking, frame-proof codes, traceability schemes**

I. Introduction

We will examine the results in the paper ”New Results on Frame-Proof Codes and Traceability Schemes” by Reihaneh Safavi-Naini and Yejing Wang [1]. Frame-proof codes were introduced in [2] and is a technique to deter from illegal copying. The basic idea is that somebody has a digital document they want to distribute to a number of users, and to make it possible to trace illegal copies, he/she incorporates small changes in the document. If a single user makes a copy of his/her document it is simple to determine the identity of the guilty user by examining the copy.

A stronger attack is if several users cooperate to create a new document that is a combination of their copies, and it is here that frame-proof codes are useful. Each user gets a copy of the digital document that corresponds to a codeword in the frame-proof code. The relation between the codeword and the document copy is that each coordinate in the code decides what alternative is chosen in one of the places where changes are allowed. It is further assumed that users working together to create an unsolicited copy can in each changeable position only choose among the alternatives given in their copies. Described in terms of words in the code space, a group of users can create any word which for every coordinate is equal to at least one of the codewords belonging to them. The combinatorial properties of a frame-proof code are such that, as long as the number of redistributors is limited, they cannot create the codeword/document of another user.

Traceability schemes were introduced in [3] and are in some ways similar to frame-proof codes. A common scenario is a broadcast of some digital data stream that is encrypted and available only to authorized users. The stream is decrypted using a decoder containing suitable decryption keys. In a traceability scheme there is a base set K of keys, of which each decoder contains a unique subset of size k. A set of users may want to create a pirate decoder by using a suitable combination of some of the keys in their decoders. In a traceability scheme, if the number of users working together is limited, any such created pirate decoder will be possible to trace to at least one of the guilty users. The idea is that this property will deter from creating pirate decoders.

In the discussed paper constant-weight codes are used to make bounds and explicit constructions. These codes have length l, constant weight w, minimum Hamming distance 2δ, and c is the number of cooperating, copy-distributing users. $H(x)$ denotes the binary entropy function, and logarithms are in base 2. We will discuss Theorems 6, 7, 10, and 11 from [1].

II. On Theorem 6

Our initial concern is Theorem 6 on binary frame-proof codes. It depends on the following displayed inequalities:

$$\frac{\log l}{l} < \sigma \quad \text{and} \quad l > \left(13 + \sqrt{13^2 + 48\sigma}\right)/12\sigma. \qquad ([1]:6)$$

We quote Theorem 6 from [1]:

Theorem 6: Let q be a prime power. Suppose there exists a c-frame-proof code with length $l \leq q$, constant weight w, and $c = l/w$. Then, for any $\sigma > 0$ and l satisfying ([1]:6), the maximum number of codewords n satisfies

$$n > \frac{1}{q^{\delta-1}} 2^{(H(\frac{1}{c})-\sigma)l}. \qquad ([1]:13)$$

There is no formal proof of this in [1], but we have studied the discussion leading to Theorem 6 in some detail to reconstruct a proof. We will not repeat the necessary steps here, but only mention that the implication in Lemma 3 of [1] is needed in the reverse direction for the proof to go through. That this implication is not an equivalence can be seen by using, for example, the code $G = \{0011, 0110, 1100\}$ in Lemma 3, see [1].

In any case, the following counterexample shows explicitly that Theorem 6 does not hold. We restate the upper bound from [4][1] on the number of codewords in a c-frame-proof code over

J.-Å. Larsson is with the Department of Mathematics, Linköping University, SE-581 83 Linköping, Sweden (e-mail: jalar@mai.liu.se).

J. Löfvenberg did this work while at the Department of Electrical Engineering, Linköping University, SE-581 83 Linköping, Sweden. J. Löfvenberg is presently with the Department of Systems Development and IT-security, FOI: Swedish Defence Research Agency, SE-164 90 Stockholm, Sweden (e-mail: jaclof@foi.se).

[1] The bound $n \leq s^{\lceil l/c \rceil} + 2c - 2$ mentioned in [1] is not valid for c-frame-proof codes as the authors claim, see [4]. The bound (1) used here is less restrictive but will still be contradicted by Theorem 6. We also note that the definitions of *feasible set* (called set of descendants in [4]) differ between [1] and [4], but it is easy to verify that in spite of this, the definitions of *frame-proof code* are equivalent.

an alphabet of size s,

$$n \leq c\left(s^{\lceil \frac{l}{c} \rceil} - 1\right). \tag{1}$$

We will compare these bounds for code length $l = 64$ and $c = 2$. In Theorem 6 we also need values for q, δ, w and σ. From the relation $c = l/w$ we obtain $w = 32$, and we will use $q = 64$. Nothing is said about δ, but the lemmas leading to Theorem 6 require that $\delta \geq 3$, so we will use $\delta = 3$. The value for σ must meet the inequalities in ([1]:6), and we use $\sigma = 7/64$.

To be in compliance with the prerequisites of Theorem 6 we must show that there exists a 2-frame-proof code with these parameters. Let I be the unity matrix of size 3, and let $\overline{0}$ and $\overline{1}$ be three-dimensional column vectors of zeros and ones respectively. Let Γ be a code with the code matrix

$$\Gamma = I^3 \overline{1}^{29} \overline{0}^{26}, \tag{2}$$

meaning the concatenation of three unity matrices, 29 one-vectors and 26 zero-vectors. This code has three codewords and is a 2-frame-proof code.

Now everything is in place, and with the proposed values in Theorem 6 we get

$$n > \frac{1}{(2^6)^2} \times 2^{64 \times (1 - 7/64)} = 2^{45}. \tag{3}$$

Using the same l and c in (1) yields

$$n \leq 2(2^{32} - 1) < 2^{33}, \tag{4}$$

which clearly contradicts expression (3). Thus we conclude that Theorem 6 in [1] does not hold.

III. On Theorem 7

We now turn to the similar Theorem 7, which we quote from [1].

Theorem 7: Let q be a prime power. Suppose there exists a c-traceability scheme with l keys, $l \leq q$, such that there are k keys in each decoder, and $c^2 = 2l/k$. Then, for any $\sigma > 0$ and l satisfying ([1]:6), the maximum number of decoders n satisfies

$$n > \frac{1}{q^{\delta-1}} 2^{(H(\frac{1}{c^2}) - \sigma)l}. \tag{[1]:15}$$

Again, there is no formal proof, but a similar exercise as for Theorem 6 shows that Lemma 5 in [1] is used in the reverse direction. We proceed straight to the counterexample. We restate from [5] an upper bound on the number of decoders in a c-traceability scheme, also given as expression (4) in [1]. In a c-traceability scheme it holds that

$$n \leq \binom{l}{t} \Big/ \binom{k-1}{t-1}, \tag{[1]:4}$$

where $t = \lceil \frac{k}{c} \rceil$ and k is the number of keys contained in each decoder.

Let us choose $l = 256$ and $c = 4$. In Theorem 7 we also need values for q, δ, k and σ. From the relation $c^2 = 2l/k$ we obtain $k = 32$, and similarly to Section II we choose $q = 256$, $\sigma = 9/256$, and $\delta = 3$.

Using $l = 256$ and $k = 32$ it is possible to construct a trivial traceability scheme with eight decoders, each containing 32 keys and no pair of decoders sharing any keys. This traceability scheme can handle (at least) $c = 4$ users working together to create a pirate decoder.

Again everything is in place so we can use the proposed values in expression ([1]:15), and this yields a lower bound on the maximal number of decoders as

$$n > 2^{-16} \times 2^{256 \times (H(1/16) - 9/256)} > 2^{61}. \tag{5}$$

The same l, c and k in expression ([1]:4) yields

$$n \leq \binom{256}{8} \Big/ \binom{31}{7} < 2^{28}, \tag{6}$$

which clearly contradicts expression (5). Thus we conclude that Theorem 7 in [1] does not hold.

IV. On Theorems 10 and 11

Even if Theorems 6 and 7 do not hold, there is an explicit construction underlying Theorems 10 and 11 in [1], also providing lower bounds for the number of code words n. The claim is:

Theorem 10: For a given integer $c > 1$, there exists a c-frame-proof code for which the parameters are restricted by ([1]:6),

$$\sigma = \tfrac{1}{2}\left(H\left(\tfrac{1}{c}\right) - \tfrac{1}{c}\right), \tag{[1]:17}$$

$$c = \tfrac{l}{w}, \tag{[1]:18}$$

and

$$\log l < \frac{1}{2} \cdot \frac{c^2}{c-1} \sigma. \tag{[1]:20}$$

Unfortunately, there is no way to choose the parameters so that ([1]:17), ([1]:18) and ([1]:20) are simultaneously satisfied. Furthermore, even if we fall back to the underlying construction, we find ourselves in similar difficulties.

To see this, we start by inserting ([1]:17) and ([1]:18) in ([1]:20), arriving at

$$\log wc < \frac{1}{4} \cdot \frac{c^2}{c-1}\left(H\left(\tfrac{1}{c}\right) - \tfrac{1}{c}\right). \tag{7}$$

The inequality $\ln x \leq x - 1$ gives us

$$H\left(\tfrac{1}{c}\right) \leq \frac{\log c + \log e}{c} \tag{8}$$

which inserted in (7) yields

$$\log w + \log c < \frac{1}{4} \cdot \frac{c^2}{c-1} \frac{\log c + \log e - 1}{c}. \tag{9}$$

The required integer $c > 1$ makes $c/(c-1) \leq 2$ and

$$\log w < \frac{1}{2}(\log e - 1 - \log c) < 0. \tag{10}$$

That is, Theorem 10 enforces weight $w < 1$. We can only conclude that the theorem is invalid as it stands in [1].

We will now go into more detail in the proof of Theorem 10, to show that also the underlying construction scheme fails. In place of ([1]:20), this construction uses a more complicated expression: the parameters must allow the existence of an integer $\delta > 0$ such that

$$\left(1-\tfrac{1}{c}\right)w-1<\delta\leq\left(H\left(\tfrac{1}{c}\right)-\tfrac{1}{c}-\sigma\right)\frac{l}{\log l}. \qquad \text{([1]:19)}$$

The left-hand inequality of ([1]:19) ensures that the code is a frame-proof code, while the right-hand inequality ensures the desired behavior of the number of codewords $n > 2^{l/c}$. The integer δ is to be used in Theorem 8 and 9 of [1] to show *existence* of a code with the desired properties. The inequality ([1]:20) is claimed to guarantee existence of such a δ, but there is no motivation of this claim in [1]. We will perform a more thourogh examination here which will show that no parameter values allow existence of such an integer $\delta > 0$.

Inserting the conditions ([1]:17) and ([1]:18) in ([1]:19) we arrive at

$$\left(1-\tfrac{1}{c}\right)w-1<\delta\leq\tfrac{1}{2}\left(H\left(\tfrac{1}{c}\right)-\tfrac{1}{c}\right)\frac{wc}{\log wc}, \qquad (11)$$

For the needed $\delta > 0$ to exist, it is clear that the following needs to be positive:

$$f(w,c)=\tfrac{1}{2}\left(H\left(\tfrac{1}{c}\right)-\tfrac{1}{c}\right)\frac{wc}{\log wc}-\left[\left(1-\tfrac{1}{c}\right)w-1\right], \quad (12)$$

Using (8), we obtain

$$f(w,c)\leq\tfrac{1}{2}(\log c+\log e-1)\frac{w}{\log wc}-\left(1-\tfrac{1}{c}\right)w+1. \quad (13)$$

This is positive if $w = 1$. When $w \geq 2 > e/2$, we have $\log w > \log e - 1$, so that (13) simplifies to

$$f(w,c)\leq w\left(\frac{1}{w}+\frac{1}{c}-\frac{1}{2}\right). \qquad (14)$$

Clearly, $0 < f(w,c)$ needs either $w = 1$ or $w = 2$ or $c = 2$, or that (c,w) is one of $(3,3)$, $(4,3)$, $(5,3)$, $(3,4)$ or $(3,5)$. We analyze these four cases separately:

a) When $w = 1$, using (8), the right-hand inequality of (11) simplifies to

$$\delta\leq\tfrac{1}{2}\left(1+\frac{\log e-1}{\log c}\right)<1. \qquad (15)$$

Consequently no such integer $\delta > 0$ exists.

b) When $w = 2$, the expression (13) is

$$f(2,c)\leq\frac{\log c+\log e-1}{\log c+1}-1+\frac{2}{c}=\frac{\log e-2}{\log c+1}+\frac{2}{c}, \qquad (16)$$

which is positive for $c = 2$, decreases to a negative minimum and then increases to 0 as c tends to infinity; it is positive only for $c < 19$, and for these values the left-hand side of (11) is also less than 1.

c) When $c = 2$ we have $H(1/2) = 1$, so that

$$f(w,2)=\frac{w}{2\log 2w}-\frac{w}{2}+1. \qquad (17)$$

This is strictly decreasing, and positive only for $w = 2$ and $w = 3$; for these values the right-hand side of (11) is also less than 1.

d) The remaining candidates $(w,c) = (3,3)$, $(4,3)$, $(5,3)$, $(3,4)$, and $(3,5)$ all give $f(w,c)$ a negative value.

That is, there is no combination of w and $c > 1$ that allows an integer $\delta > 0$ obeying expression ([1]:19). A little more effort will show that for some combinations of w and c, there are $\delta > 0$ that obey *either* the left or the right inequality, but not both. For these values of δ, the constructed codes are *either* guaranteed to be c-frame-proof codes *or* guaranteed to have a number of code words $n > 2^{l/c}$, but the constructed codes are *never* guaranteed to have *both* properties.

Similar reasoning holds for Theorem 11, with the sole difference that the parameter c is inserted squared in the equivalent of expressions ([1]:20) and ([1]:19). And in a similar fashion, the construction does not allow construction of a traceability scheme that is guaranteed to have the desired number of decoders.

V. Conclusions

We have found that Theorems 6 and 7 of [1] for some choices of parameter values violate previously published upper bounds. They can clearly not be valid as they stand.

We have also found that Theorems 10 and 11 of [1] are invalid. There are no parameter values that fulfill the given bounds simultaneously; the requirements are simply too restrictive. In other words, the theorems cannot be used to prove existence of codes with the desired properties. The underlying construction gives codes which may have one of the two desired properties, but the codes are never guaranteed to have both.

References

[1] R. Safavi-Naini and Y. Wang, “New results on frame-proof codes and traceability schemes,” *IEEE Trans. Inform. Theory*, vol. 47, no. 7, pp. 3029–3033, Nov. 2001.

[2] D. Boneh and J. Shaw, “Collusion-secure fingerprinting for digital data,” in *Advances in cryptology — CRYPTO’95*, 1995, pp. 452–465.

[3] B. Chor, A. Fiat, and M. Naor, “Tracing traitors,” in *Advances in cryptology — CRYPTO’94*, 1994, pp. 257–270.

[4] J. N. Staddon, D. R. Stinson, and R. Wei, “Combinatorial properties of frameproof and traceability codes,” *IEEE Trans. Inform. Theory*, vol. 47, no. 3, pp. 1042–1049, Mar. 2001.

[5] D. R. Stinson and R. Wei, “Combinatorial properties and constructions of traceability schemes and frameproof codes,” *SIAM J. of Discrete Math.*, vol. 11, pp. 41–53, Feb. 1998.